# NVM Express Technical Errata

| Errata ID | 014 |
|---|---|
| Change Date | 6/13/2011 |
| Affected Spec Ver. | NVM Express 1.0a |
| Corrected Spec Ver. | |

Submission info

| Name | Company | Date |
|---|---|---|
| Darren Lasko | Intel | 5/31/2011 |
| Kevin Marks | Dell | 5/31/2011 |
| Kwok Kong | IDT | 5/31/2011 |
| Eric Peterson | Expert IO | 5/31/2011 |

This erratum modifies the Security Send and Security Receive commands to utilize 32-bit Transfer Length / Allocation Length fields to align with corresponding TCG material.

This erratum corrects inconsistencies in the Set Features optional/mandatory requirements.

This erratum clarifies that PRP Lists shall be minimally sized.

This erratum makes editorial changes and clarifications in section 7.

This erratum clarifies the value read from a deallocated LBA's metadata.

Description of the specification technical flaw

---

*Modify the third paragraph of section 5.14 as shown:*

Each Security Receive command ~~shall~~ returns the appropriate data corresponding to a Security Send command as defined by the rules of the Security Protocol.  The Security Receive command data may not be retained if there is a loss of communication between the controller and host, or if a controller reset occurs.

*Modify Figure 90 as shown below:*

**Figure 90: Security Receive – PRP Entry 1**

| Bit | Description |
|---|---|
| 63:00 | **PRP Entry 1 (PRP1):** ~~Indicates~~ Specifies a data buffer that contains the security protocol information.  The buffer shall not have more than one physical discontinuity and shall be 4KB minimum in size. |

*Modify Figure 92 as shown below:*

**Figure 92: Security Receive – Command Dword 10**

| Bit | Description |
|---|---|
| 31:24 | **Security Protocol (SECP):** This field ~~indicates~~ specifies the security protocol as defined in SPC-4.  The controller shall fail the command with Invalid Parameter indicated if an unsupported value of the Security Protocol is specified. |
| 23:08 | **SP Specific (SPSP):** The value of this field is specific to the Security Protocol as defined in SPC-4. |
| 07:00 | Reserved |

*Modify Figure 93 as shown below:*

**Figure 93: Security Receive – Command Dword 11**

| Bit | Description |
|---|---|
| 31:00 ~~31:16~~ | **Allocation ~~Transfer~~ Length (AL ~~TL~~):** The value of this field is specific to the Security Protocol as defined in SPC-4. |
| ~~15:00~~ | ~~Reserved~~ |

*Modify section 5.14.1 as shown below:*

~~When~~ If the command is completed, then the controller shall post a completion queue entry to the Admin Completion Queue indicating the status for the command.

*Remove sections 5.14.2 and 5.14.3 as shown below:*

~~5.14.2   Security Protocol~~

~~The Security Protocol field for the Security Receive command identifies the security protocol as defined in SPC-4.~~

~~5.14.3   Transfer Length~~

~~The value of the Transfer Length is defined by the Security Protocol.~~

*Modify section 5.14.4 as shown below:*

A Security Receive command with the Security Protocol field set to 00h shall return information about the security protocols supported by the controller. This command is used in the security discovery process and is ~~shall~~ not ~~be~~ associated with a Security Send command.  Refer to SPC-4 for the details of Security Protocol 00h and the SP Specific field.

*Modify the first paragraph of section 5.15 as shown below:*

The Security Send command is used to ~~specify~~ transfer security protocol data to the controller. The data structure transferred to the controller as part of this command contains security protocol specific commands to be performed by the controller. The data structure transferred may also contain data or parameters associated with the security protocol commands.  Status and data that is to be returned to the host for the security protocol commands issued by Security Send command are retrieved with the Security Receive command defined in section 5.14.

*Modify Figure 97 as shown below:*

**Figure 94: Security Send – PRP Entry 1**

| Bit | Description |
|---|---|
| 63:00 | **PRP Entry 1 (PRP1):** ~~Indicates~~ Specifies a data buffer that contains the security protocol information.  The buffer shall not have more than one physical discontinuity and shall be 4KB minimum in size. |

*Modify Figure 96 as shown below:*

**Figure 96: Security Send – Command Dword 10**

| Bit | Description |
|---|---|
| 31:24 | **Security Protocol (SECP):** This field ~~indicates~~ specifies the security protocol as defined in SPC-4.  The controller shall fail the command with Invalid Parameter indicated if a reserved value of the Security Protocol is specified. |
| 23:08 | **SP Specific (SPSP):** The value of this field is specific to the Security Protocol as defined in SPC-4. |
| 07:00 | Reserved |

*Modify Figure 97 as shown below:*

**Figure 97: Security Send – Command Dword 11**

| Bit | Description |
|---|---|
| 31:00 ~~31:16~~ | **Transfer Length (TL):** The value of this field is specific to the Security Protocol as defined in SPC-4. |
| ~~15:00~~ | ~~Reserved~~ |

*Modify section 5.15.1 as shown below:*

~~When~~ If the command is completed, then the controller shall post a completion queue entry to the Admin Completion Queue indicating the status for the command.

*Remove sections 5.15.2 and 5.15.3 as shown below:*

~~**5.14.2 Security Protocol**~~

~~The Security Protocol field for the Security Send command identifies the security protocol as defined in SPC-4.~~

~~**5.14.3 Transfer Length**~~

~~The value of the Transfer Length is defined by the Security Protocol.~~

*Modify the first three paragraphs and heading of section 5.12.1.3 as shown below:*

**5.12.1.3 LBA Range Type (Feature Identifier 03h)~~, (Optional)~~**

This feature indicates the type and attributes of LBA ranges that are part of the specified namespace. The LBA range information may be used by a driver to determine if it may utilize a particular LBA range; the information is not intended to be exposed to higher level host software. This information is for use by host software; it is not interpreted by the controller.

This is optional information that is not required for proper behavior of the system. However, it may be utilized to avoid unintended host software issues. For example, if the LBA range indicates that it is a RAID volume then a driver that does not have RAID functionality should not utilize that LBA range (including not overwriting the LBA range). The optional information may be utilized by the driver to determine whether the LBA Range should be exposed to higher level host software.

The LBA Range Type uses Command Dword 11 and specifies the type and attribute information in the data structure indicated in Figure 77. The data structure is 4096 bytes in size ~~and shall be physically contiguous~~.

*Modify Figure 76 as shown below:*

**Figure 76: LBA Range Type – Command Dword 11**

| Bit | Description |
|---|---|
| 31:06 | Reserved |
| 05:00 | **Number of LBA Ranges (NUM):** This field ~~indicates~~ specifies the number of LBA ranges specified in this command. This is a 0's based value. |

*Modify Figure 77 as shown below:*

**Figure 77: LBA Range Type – Data Structure Entry**

| Byte | Description |
|---|---|
| 00 | **Type (Type):** ~~Identifies~~ Specifies the Type of the LBA range.  The Types are listed below.<br><br>| Value | Description |<br>\|---\|---\|<br>\| 00h \| Reserved \|<br>\| 01h \| Filesystem \|<br>\| 02h \| RAID \|<br>\| 03h \| Cache \|<br>\| 04h \| Page / swap file \|<br>\| 05h – 7Fh \| Reserved \|<br>\| 80h - FFh \| Vendor Specific \| |
| 01 | **Attributes:** ~~Identifies~~ Specifies attributes of the LBA range.  Each bit defines an attribute.<br><br>| Bit | Description |<br>\|---\|---\|<br>\| 0 \| If set to '1', the LBA range may be overwritten.  If cleared to '0', the area should not be overwritten. \|<br>\| 1 \| If set to '1', the LBA range should be hidden from the OS / EFI / BIOS.  If cleared to '0', the area should be visible to the OS / EFI / BIOS. \|<br>\| 2 – 7 \| Reserved \| |
| 15:02 | Reserved |
| 23:16 | **Starting LBA (SLBA):** This field ~~indicates~~ specifies the 64-bit address of the first ~~LBA~~ logical block that is part of this LBA range. |
| 31:24 | **Number of Logical Blocks (NLB):** This field ~~indicates~~ specifies the number of logical blocks that are part of this LBA range.  This is a 0's based value. |
| 47:32 | **Unique Identifier (GUID):** This field is a global unique identifier that uniquely ~~identifies~~ specifies the type of this LBA range.  Well known Types may be defined and are published on the NVM Express website. |
| 63:48 | Reserved |

*Modify the heading of section 5.12.1.6 as shown below:*

**5.12.1.6 Volatile Write Cache (Feature Identifier 06h), (Optional)**

*Add the following paragraph as the last paragraph of section 4.3 as shown below:*

PRP Lists shall be minimally sized with packed entries starting with entry 0.  If more PRP List pages are required, then the last entry of the PRP List page is a pointer to the next PRP List page.  The total number of PRP entries is implied by the command parameters and memory page size.

***Modify the first paragraph of section 7.4.1 as shown below:***

To allocate I/O Submission Queues and I/O Completion Queues, host software follows these steps:

1. ~~Software allocates~~ Allocates the Admin Submission and Completion Queues by configuring the Admin Queue Attributes (AQA), Admin Submission Queue Base Address (ASQ), and Admin Completion Queue Base Address (ACQ) registers appropriately.
2. ~~Software issues~~ Issues a Set Features command for the Number of Queues attribute in order to request the number of I/O Submission Queues and I/O Completion Queues desired.  The completion of this Set Features command indicates the number of I/O Submission Queues and I/O Completion Queues allocated.
3. ~~Software determines~~ Determines the maximum number of entries supported per queue (CAP.MQES) and whether the queues are required to be physically contiguous (CAP.CQR).
4. ~~Software allocates~~ Allocates the desired I/O Completion Queues within the limitations of the number allocated by the controller and the queue attributes supported (maximum entries and physically contiguous requirements) by using the Create I/O Completion Queue command.
5. ~~Software allocates~~ Allocates the desired I/O Submission Queues within the limitations of the number allocated by the controller and the queue attributes supported (maximum entries and physically contiguous requirements) by using the Create I/O Submission Queue command.

***Modify the first paragraph of section 7.4.2 as shown below:***

There is one Admin queue pair associated with multiple I/O queue pairs.  The Admin Submission Queue and Completion Queue~~s~~ are used to carry out functions that impact the entire controller.  An I/O Submission Queue and Completion Queue may be used to carry out I/O (read/write) operations and may be distributed across CPU cores and threads.

***Modify section 7.4.3 as shown below:***

To abort a large number of commands ~~previously issued to an I/O Submission Queue~~, the recommended procedure is to ~~abort all commands issued to that I/O Submission Queue~~ delete and recreate the I/O Submission Queue.  ~~To abort all commands for a Submission Queue, the queue should be deleted and recreated.~~  Specifically, to abort all commands that are issued to the Submission Queue host software should issue a Delete I/O Submission Queue command for that queue.  After the queue has been successfully deleted, indicating that all commands have been aborted, then host software should recreate the queue by issuing a Create I/O Submission Queue command.  Host software ~~Software~~ may then re-issue any commands desired to the associated I/O Submission Queue.

***Modify step 3 in the initialization actions in section 7.6.1 as shown below:***

3. The controller settings should be configured.  Specifically:
    a. The arbitration mechanism should be selected in CC.AMS.
    b. The memory page size should be initialized in CC.MPS.
    c. The I/O ~~command set~~ Command Set that is to be used should be selected in CC.CSS.

***Modify step 10 in the initialization actions in section 7.6.1 as shown below:***

10. If the host desires asynchronous notification of error or health events, the host should issue an appropriate number of Asynchronous Event Request commands.  This step may be done at any point after the controller signals it is ready (i.e., CSTS.RDY is set to '1').

*Modify the first paragraph of section 7.6.1.1 as shown below:*

The Software Progress Marker feature, defined in section 5.12.1.12, indicates the number of times pre-boot software has loaded prior to the OS successfully loading.  If the pre-boot software load count becomes large, it may indicate there are issues with cached data within the NVM since the OS driver software has not set this field to 0h recently.  In this case, the OS driver software may choose to use the NVM more conservatively (e.g., not utilize cached data).

*Modify the last four paragraphs of section 7.6.2 as shown below:*

It is recommended that the host wait a minimum of one second for the shutdown operations to complete.  It is not recommended to disable the controller via the CC.EN field.  This causes a controller reset condition which may impact the time required to complete shutdown processing.

To start executing commands on the controller after a shutdown operation, a reset (CC.EN cleared from '1' to '0') is required.  The initialization sequence should then be executed.

It is an implementation choice whether the host aborts all outstanding commands to the Admin Queue prior to the shutdown.  The only commands that should be outstanding to the Admin Queue at shutdown are Asynchronous Event Request commands.

~~It is not recommended to disable the controller via the CC.EN field.  This causes a controller reset condition which may impact the time required to complete shutdown processing.~~

*Modify the first paragraph of section 6.6.1.1 as shown below:*

An LBA that has been deallocated using the Dataset Management command is no longer deallocated when the LBA is written.  Read operations do not affect the deallocation status of an LBA.  The value read from a deallocated LBA shall be deterministic; specifically, the value returned by subsequent reads of that LBA shall be the same until a write occurs to that LBA.  The values read from a deallocated LBA and its metadata (excluding protection information) shall be all zeros, all ones, or the last data written to the associated LBA and its metadata.  The values read from an unwritten or deallocated LBA's protection information field shall be all ones (indicating the protection information shall not be checked).

Disposition log

| | |
|---|---|
| 5/31/2011 | Erratum captured. |
| 6/8/2011 | Updated protection information material. |
| 6/13/2011 | Updated LBA Range Type, PRP List description, and unwritten Protection Information. |
| 7/21/2011 | Erratum ratified. |

*Technical input submitted to the NVMHCI Workgroup is subject to the terms of the NVMHCI Contributor's agreement.*